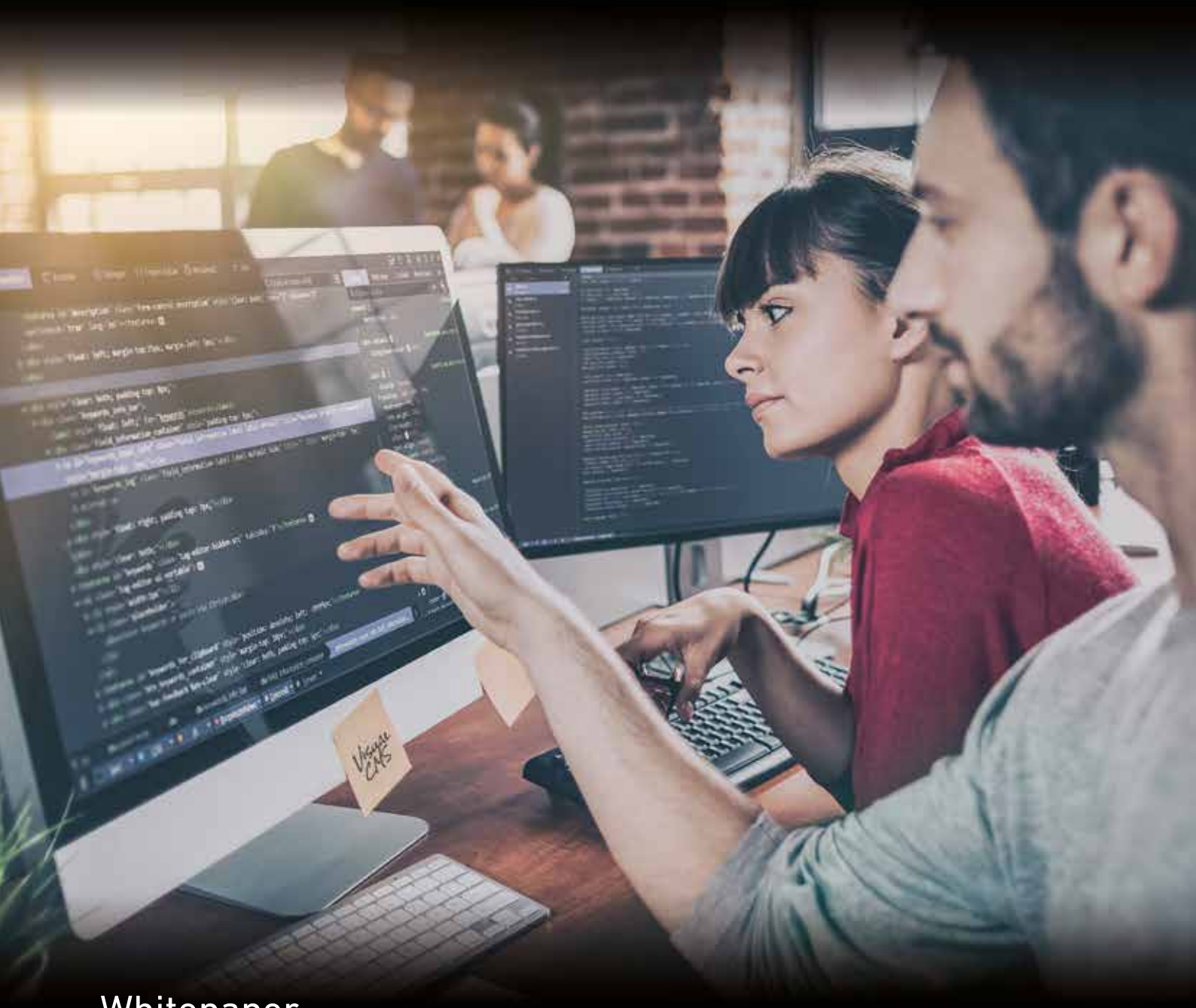
OXID
esales



Whitepaper

SECURITY ISSUE AT OXID? OUR PROCESS!

oxid-esales.com

CONTENTS

Introduction 3

1. A security issue in OXID eShop is ... 3
2. How do such security issues occur? 3
3. What is at stake? 4
4. Sensitisation 4
5. How do these reports get to us? 4
6. The Security Team 5
7. Internal organisational structure and distribution of information 5
8. First assessments 5
9. Different processes according to the traffic light principle 6
   - 9.1 Code:green 6
   - 9.2 Code:yellow 6
   - 9.3 Code:red 7
10. Modules and extensions 8
11. Staying up-to-date with security issues 8
12. Documentation of previous security issues 8
13. Rework 8

Contact 8

As a manufacturer of software which is partly distributed with a commercial licence but also with an Open Source licence, we are duty-bound to our customers and users to deal with reported security issues very carefully.

We are aware that shop software, in particular, can be a business-critical application, and incorrect handling, not treating information confidentially or information at the wrong time can lead to financial loss or damage to the company's public image. That is why we insist on being in control of the communication process ourselves at all times.

1 A SECURITY ISSUE IN OXID ESHOP IS ...

a security gap that can be exploited by an attacker to

+ disrupt the flawless execution of the application (e.g. DoS/Denial of Service),
+ steal data without authorisation (e.g. "scamming"),
+ cause financial damage to the distributor (e.g. to gain personal advantages during a purchase),
+ gain unauthorised access to the application or the server, e.g. to plant malicious code in the application (e.g. via XSS, exploiting as a spambot, hosting of forbidden contents etc.).

2 HOW DO SUCH SECURITY ISSUES OCCUR?

No software is free from errors, so-called "software bugs". Of course, this also applies to security-related software errors that occur accidentally. Even the best training for completely secure programming will not be able to eradicate human aspects such as "overlooking", "inattentiveness at that moment" or errors from years ago when people wouldn't have even imagined a chance for exploitation. That means such "failures" are almost commonplace in all systems.

Nobody likes to admit this because it would be seen as a weakness. But if we deal with these errors progressively, we can prevent worse damage. We force ourselves to do this by publishing the software core as an Open Source edition, knowing full well that it can be tested thoroughly by complete strangers.

3 WHAT IS AT STAKE?

Open Source development is publicly accessible. That's why security issues can be located right in the code. Based on the huge numbers of LoC (lines of code), this is not particularly easy, which is why attackers focus on changes in code, so-called commits.

If there is a security issue that is exploited before it can be reported to the manufacturer (to us in this case), we talk about a so-called **"Zero Day Exploit"**, which has so far only happened once in the OXID universe.

Depending on the severity of the attack, the chances are that the reputation of the online distributor as well as of the partner agency or of OXID eSales AG will be affected. In the worst case scenario, this can escalate to a full-blown PR crisis which can cause enormous damage to the company's public image. Of course, we want to avoid this at all costs.

4 SENSITISATION

Corrections that resolve security issues are provided as updates. Workarounds can also often be made available which (adapted accordingly) can still be used in versions that are no longer supported. "Often" means "not always", which is why we request that customers, users and agencies on all channels work as close to the current release as possible. Updates and upgrades have to be included in the initial planning and in the regular running budget for software maintenance by the shop operator, in particular.

Data privacy specialists (GDPR) as well as the BSI (German Federal Office for Information Security) do not ask users without reason to keep software up-to-date. Arguments such as "I'm too small for that" or "who can my data be of use to" are fallacies!

5 HOW DO THESE REPORTS GET TO US?

We regularly undergo internal security audits as well as those carried out by external agencies. It is interesting to note that even in this case, not all potential weaknesses can always be revealed. In the past, potential security issues have been reported by developers but also from the partner and customer landscape. We also receive reports from completely external sources such as from security officers in customer projects, research teams at universities all over the world who seem to randomly check out Open Source projects.

6 THE SECURITY TEAM

The Security Team is currently made up of up to three positions:
the Head of Development Manager, a Senior Technical Lead and the Software Support Engineer. The set-up of the team has not been chosen at random. It offers various advantages: The process can be scaled and even works if one team member is not available. The team is just the right size allowing it to make decisions quickly without many complications. Another aspect, which must not be underestimated psychologically: The responsibility must not remain solely with one person.

7 INTERNAL ORGANISATIONAL STRUCTURE AND DISTRIBUTION OF INFORMATION

The email address security@oxid-esales.com is an internal distributor. The following groups are notified:

+ The **Security Team** will respond immediately to an incoming email with the information that we will take care of assessing the potential security issue and subsequent steps as quickly as possible.
+ The **Product Development Team**, which will deal with the potential security issue without delay. The assessment whether this is a reproducible security issue, what damage can potentially be caused and how and with what effort the error can be corrected, is forwarded to the Security Team.
+ The **Support Team** to stay informed and, if applicable, to lend support.

8 FIRST ASSESSMENTS

After the developers' assessment, the Security Team calculates the so-called CVSS, a standard score with a scale from 0 to 10 to judge the severity of the security issue, and discusses the next steps: When will the bug be fixed, when will who be notified and when will the closed security gap be published as a release? The CVSS is an important indicator of this assessment: One can't simply rely on human decision-making because this can be affected by various factors (lack of time, the person's frame of mind on the day etc.).

Depending on the severity, three different processes can now be triggered:

+ CVSS < 3 = **Code:green**,
+ CVSS > 3 and < 7 = **Code:yellow** and
+ CVSS > 7 means **Code:red**.

The person who reported the security issue will then be notified of the planned procedure and obligated to maintain confidentiality until the release is published.

9 DIFFERENT PROCESSES ACCORDING TO THE TRAFFIC LIGHT PRINCIPLE

On the basis of the CVSS assessment, different processes are used which are explained here in detail:

9.1 CODE:GREEN

For security issues with CVSS < 3, a potential attacker has to overcome huge obstacles to cause relatively little damage - in other words, it's not worth it. These are often attack opportunities that are classified as CSFR (Cross Site Forgery Request).

+ A CVE number (see below) is not requested.
+ A security bulletin is not created.
+ The bug tracking entry remains open, the bug is processed with high priority and is included in the next planned release.
+ The correction (fix) takes place in the open GitHub repository.
+ OXID employees do not receive an internal notification.
+ Advance information to partners, customers and NDA signatories is not sent.
+ In the release notes, there is a mention with reference to the bug tracking entry that a "security improvement" is contained in the release.

9.2 CODE:YELLOW

Security issues with CVSS > 3 and < 7 are processed using 'Code:yellow'. In a typical attack scenario, missing form validations are exploited, for example. Some effort is required and damage can be caused.

+ A CVE number is requested from MITRE.
+ An OXID security bulletin is created and published in the online documentation, password-protected.
+ In addition to a security bulletin, a bilingual FAQ is created on the online documentation, which is publicly available at the latest from the publication of the security bulletin. This FAQ is primarily aimed at a readership that is not proficient in reading security bulletins and serves as a "landing page" for further publications (see below).
+ The correction takes place in a closed GitHub repository which is combined with the public GitHub repository just before the release.
+ All employees with customer contact receive an exact schedule stating when partners, customers and NDA signatories will be notified, when the OXID eShop release will be available and when the security bulletin will be published.
+ Usually ten to 14 days before the release, an advance notification is sent to partners, customers and NDA signatories with the passwords for the security bulletin (which may contain a workaround) and information about the schedule for the next steps. In this notification, we ask for confidentiality until the release. If one of the recipients does not comply, he or she will be blocked as of the next receipt of such information.
+ The public release of the OXID eShop Suite follows, which contains a fix. The release notes include a clear reference to resolved security issues and the request for a quick update.
+ About ten days after this release, the security bulletin is published and the status of the bug tracking entry is changed to "public" (if the Community Edition has also been affected).
+ In addition to MITRE, various other security databases are also notified.

9.3 CODE:RED

A Code:red is a security issue with a calculated CVSS > 7. Examples: API access, which is completely open from outside or the opportunity to get admin access via the front end of the shop. The effort to stage such an attack is often minimal. Maximum damage can be caused. That is why the process is essentially based on the same steps as described in "Code:yellow", but differs in the following procedures:

+ We reserve the right to publish an OXID eShop release outside the usual release cycles (quarterly) in this case.
+ In addition to a security bulletin, a bilingual FAQ is created in the OXID forge which will be publicly accessible at the latest from publication of the security bulletin. This FAQ is mainly designed for readers who can't understand security bulletins. It is used as a "landing page" for further publications (see below).
+ The timing of the advance notification of NDA signatories, partners and customers mentioned above is different for each group: First, NDA signatories are notified, and a few days later, partner agencies and customers without NDA. This procedure is designed to ensure that partners are notified of these processes and schedules before they are confronted with the fact by their customers, without knowing anything about it.
+ We provide hosting providers (not only hosting partners!) with rules for the mod_security Apache module. We obtain a mailing list for this from our co-operation with the SIWECOS project, the CMS Garden and friendly Open Source CMS and shop systems. These rules enable the hosting provider as the server owner to forward relevant web server enquiries directly to the Nirvana to dispense with any support effort in the event of an attack.
+ We will notify the BSI (German Federal Office for Information Security) as well as specialist press (e.g. heise.de, golem.de etc.) to reach the biggest recipient group possible and users of OXID eShop, in particular, and to get these to update their software.
+ As opposed to the Code:yellow procedure, the security bulletin as well as the bug tracking entry are published together with the release of a new OXID eShop version because it may only be hours before the first attack.

10 MODULES AND EXTENSIONS

Modules and extensions supplied with the OXID eShop Suite are included in the processes mentioned above. We keep close contact with the manufacturers of these modules to guarantee a smooth information flow and integration into the suite.

Modules that are not supplied with the OXID eShop Suite cannot be dealt with using this process. We recommend direct contact with the manufacturer.

11 STAYING UP-TO-DATE WITH SECURITY ISSUES

There are various options for staying up-to-date with security issues:

+ Partners, customers and NDA signatories are notified in advance by email (see above).
+ New releases are published via release notes and contain information that security vulnerabilities have been fixed.

12 DOCUMENTATION OF PREVIOUS SECURITY ISSUES

Previous security bulletins are filed here: **https://oxidforge.org/en/security#previous_security_information** or can be traced using the RSS feed above.

13 REWORK

After each case has been closed, the Security Team gets together again for a rework and to discuss what went well and what did not go well, where improvements should be made in the process or which decisions must be made elsewhere in the company. We frequently ask partners and customers about this, meaning we obtain valuable feedback that we can use for the next case.

CONTACT

If you have any questions about the security process or comments, please contact the Security Team in English at **security@oxid-esales.com** or visit **https://security.oxid-esales.com.**.

OXID eSales AG
Bertoldstraße 48
79098 Freiburg, Germany

Phone: +49 761 368 89 0
Fax:+49 761 368 89 29
info@oxid-esales.com

www.oxid-esales.com
www.twitter.com/OXID_eSales
www.facebook.com/OXID.eSales